

<b>3ªEVALUACIÓN</b>	<b>Curso:</b> <b>1ºDAM/DAW</b>
<b>Nombre:</b>	<b>Fecha: 19/5/2022</b>

**LEE ATENTAMENTE LAS INSTRUCCIONES DEL EXAMEN. ES OBLIGATORIO REALIZAR Y ENTREGAR EL EXAMEN SEGÚN LAS MISMAS.**

**Entrega:** El examen se entregará en la tarea creada para tal fin en el aula virtual. Será entregado en formato **zip** (sólo se aceptará este formato en la entrega) con la siguiente nomenclatura: **APELLIDO\_NOMBRE\_EX3**. Dentro del zip, estarán cada uno de los ejercicios.

**¡¡¡OJO!!! MUCHO CUIDADO CON LA HORA DE LA TAREA DEL AULA VIRTUAL. LLEGADA LA HORA, LA TAREA NO PERMITE SUBIR ARCHIVOS, POR LO QUE AQUELLOS EXÁMENES QUE NO HAYAN SIDO SUBIDOS, NO SERÁN CORREGIDOS.**

**Calificación:** Los ejercicios **sólo** se empezarán a corregir si **compilan** y **ejecutan correctamente**. Para ello, no puede haber: ni errores sintácticos, ni bucles infinitos.

TODO EXAMEN QUE SEA COPIADO DE OTRO COMPAÑERO, SERÁ SUSPENSO PARA AMBOS. TODO EXAMEN COPIADO DE CUALQUIER OTRA FUENTE, SERÁ SUSPENSO.

Se podrán consultar los apuntes de clase, los cuales se entregarán junto al material del examen. Durante la realización del examen se desconectará el acceso a internet. Únicamente se permitirá durante el inicio del examen y cuando el alumno vaya a realizar la entrega.

**Duración: 210 minutos (3 horas y media)**

### Ejercicio 1: (4 puntos)

El departamento de I+D de una fábrica de electrodomésticos ha propuesto el desarrollo de un prototipo software que permita simular una pequeña parte de un robot de cocina inteligente. En este caso en concreto, el sistema a probar estaría relacionado con el dispensador de ingredientes que empleará el robot para la elaboración de las recetas.

Los requisitos que se proponen para el sistema son los siguientes:

El robot será capaz de cocinar cualquier tipo de receta. El robot internamente constará de un sistema de cartuchos que permitirá al sistema disponer de una especie de despensa para la elaboración de las recetas.

Sólo se permite un tipo de cartucho para cada ingrediente (no se pueden repetir).

Los ingredientes vienen almacenados en cartuchos no recargables que permiten consultar el ingrediente y la cantidad que contiene. Existen dos tipos de cartuchos, cartuchos finitos (tienen una cantidad fija) y cartuchos infinitos (no se gastan). Independientemente del tipo de cartucho del que se trate, este permite dispensar de forma automática una cantidad dada, pudiendo informar al sistema en caso no disponer de la cantidad solicitada mediante una excepción de tipo `CantidadNoDisponible`, o en caso de llegar a agotarse mediante una excepción de tipo `Vacio`.

Cada vez que se mande al robot la orden de elaborar un plato, este realizará las siguientes comprobaciones:

- El sistema comprobará que dispone de los ingredientes y cantidades necesarias que necesita el plato para su elaboración, en caso de no disponer del ingrediente, el sistema lanzará una excepción de tipo `FaltaIngrediente`, y en caso de no disponer de la cantidad necesaria, el sistema lanzará una excepción de tipo `CantidadInsuficiente`.
- En caso de superar satisfactoriamente las comprobaciones del paso anterior, el sistema procederá a actualizar las cantidades restantes en cada uno de los cartuchos mediante el método **`dispensa(cantidad)`** disponible para cualquier tipo de cartucho.
- 

El robot permite añadir o reemplazar cartuchos ya sea porque no disponen de la cantidad necesaria para la elaboración de la receta, o porque el

cartucho se ha agotado.

Al realizar el reemplazo del cartucho se ha de tener en cuenta las siguientes consideraciones:

- No se permiten repetir ingredientes.
- En caso de existir el ingrediente, se reemplazará completamente el cartucho.
- No hay límite de cartuchos.

Para facilitar la resolución del ejercicio se facilita el esqueleto de las clases necesarias para la resolución del ejercicio. Se deberá de completar el código donde aparezca `//TODO`. Igualmente se ha provisto de un programa principal (`main/Main.java`) que permitirá realizar una serie de comprobaciones básicas.

**Apartado 1 (1 puntos):** Implementa los métodos que aparecen en la clase `robot/RobotDeCocina.java`.

**Apartado 2 (1 puntos):** completa la clase o clase abstracta o interfaz (según estimes conveniente) `Cartucho.java` que se encuentra en el paquete `cartucho`.

**Apartado 3 (1 puntos):** Implementa los métodos que aparecen en la clase `DispensadorCartuchos.java` que se encuentra en el paquete `cartucho`.

**Apartado 4 (0,5 puntos):** Haz los cambios necesarios en las clases: `Agua`, `Aceite`, `Chorizo`, `Huevo`, `Sal` del paquete `cartucho`, según la elección que hayas hecho en el apartado 2

**Apartado 5 (0,5 puntos):** Genera las excepciones necesarias que requiera el programa. Es necesario que generes tanto el constructor vacío como un constructor que reciba una cadena de texto, dispones de la estructura básica de todas ellas en el paquete `cartucho`.

## Ejercicio 2: (3 puntos)

Necesitamos desarrollar el software de control de una máquina de clasificación de frutas la cual podrá clasificar la fruta según distintos criterios.

La máquina tiene capacidad "infinita" y permite también que haya repeticiones.

Para facilitar la resolución del ejercicio se facilita el esqueleto de las clases necesarias para la resolución del ejercicio. Se deberá de completar el código donde aparezca //TODO. Igualmente se ha provisto de un programa principal (main/Main.java) que permitirá realizar una serie de comprobaciones básicas.

**Apartado 1 (1.5 puntos):** Completa los métodos que aparecen en la clase Clasificadora.java, y haz los cambios que estimes necesarios para implementar Comparable en la clase Fruta.

**Apartado 2 (1.5 puntos):** Completa los métodos que aparecen en la clase Main.java. Nota: la clase **FrutaPatron** se empleará para establecer los datos de la fruta que servirá como referencia para hacer uso de Comparartor.

### Ejercicio 3: (3 puntos)

El departamento financiero de una empresa de transportes ha solicitado el desarrollo de un software el cual permita calcular los valores de conversión de una serie de divisas a otra moneda de cambio. Para realizar el procesado de los datos, la entidad entregará un fichero de texto en formato csv: divisas.txt, dicho fichero estará disponible dentro de relativa a la carpeta del proyecto del ejercicio.

La información que encontraremos en el fichero será la siguiente:

- En la primera linea se encontrará la información relativa a la moneda que hay que emplear para realizar el cambio y cuyas columnas se distribuirán de la siguiente forma:

**nombreMoneda;valorCambio respecto al Dolar americano.**

- En las siguientes lineas estarán almacenados los datos de las divisas a procesar y cuyas columnas se distribuirán de la siguiente forma:

**Nombre del país;Nombre de la moneda;Valor en dolares (en formato decimal con coma)**

Para la realización de dicho programa se pide lo siguiente:

**Apartado 1 (1.5 puntos):** Lee los datos almacenados en el fichero csv de entrada y generando las clases que estimes necesarias para representar dichos datos.

**Apartado 2 (0,5 puntos):** Guarda en un fichero binario llamado "divisas.dat" los objetos generados en el apartado 1.

**Apartado 3 (0,5 puntos):** Lee el fichero binario los datos guardados en el paso 2 y devuelve los datos almacenados

**Apartado 4 (0,5 puntos):** Genera un fichero de texto llamado divisasToString.txt cuyo contenido será el resultado de invocar el método toString() de la colección que resulta de aplicar la conversión de las divisas.

Para facilitar la resolución del ejercicio se facilita el esqueleto de las clases necesarias para la resolución del ejercicio. Se deberá de completar el código donde aparezca //TODO . Igualmente se ha provisto de un programa principal (main/Main.java) que permitirá realizar una serie de comprobaciones básicas.